# FERMILAB STOCHASTIC COOLING CODE USER'S GUIDE

John Marriner and Vladimir Visnjic
*Fermi National Accelerator Laboratory*
*Batavia, IL 60510*

## Abstract

The code used for the numerical integration of the Fokker–Planck equation for the momentum cooling of antiprotons in the Fermilab Accumulator is described and documented.

# 1. INTRODUCTION

The Stochastic Cooling Code described in this report was originally written by S. van der Meer for the CERN $\bar{p}$ accumulator AA. It was modified subsequently by John Marriner to describe Fermilab Accumulator.

The main changes introduced in the present version are:

1. The program originally was written and ran on CYBER in single precision and contained a large number of CYBER extensions. Those have been rewritten in standard FORTRAN 77 and whole program has been changed to double precision. This involved, besides rewriting all the lines that contained non standard commands, also purging a substantial number of CERNLIB subroutines which do not admit double precision and replacing them by FORTRAN code. All COMMON blocks had to be rewritten, since changing to double precision changes the storage length of the reals but not of integers. A few variables had to be kept REAL*4 since they are EQUIVALENCE-d with integers and appear in a group of subroutines ZBOOK (CERNLIB) which do not admit double precision. It would be desirable to completely eliminate all ZBOOK routines from the program and replace them by standard FORTRAN and I expect that this will be done in a future version of the program.

2. The program had a severe computational instability for some values of input parameters, which unfortunately included precisely those describing the present stochastic cooling setup at Fermilab. A typical outcome was a wildly oscillatory energy distribution of the $\bar{p}$ stack. This was cured by allowing the program that solves the Fokker–Planck equation to work with a variable size of the energy bin. The worst case was stabilized with the energy bin a factor of 4 smaller than originally. The price was an increase by about a factor of 20 in the CPU time.

3. A new graphics subroutines were added to the code, based on the HIGZ graphics package. It is pretty straightforward now to get a graphical representation of any quantity computed in the program.

In its present form, the program compiles and runs flawlessly on the AMDAHL under CMS, the VAX under VMS and on the UNIX machines. The main number crunching is presently done on the AMDAHL, where a computation of 10 hours of stacking takes about 20 min. CPU time (simplest case).

At this point it is necessary to expand the graphics capabilities of the program. Among other things, one has to be able to see the results as they are being computed which will enhance our capability to efficiently use the program. This will unavoidably limit the portability of the code since the graphics packages differ from machine to machine, but the added flexibility is worth paying the price. The strategy adopted is to use consistently the CERN libraries since they are available on all the computers at Fermilab.

The plan of this paper is as follows. In Section 2 a dictionary of most of the variables is given providing a short definition and the subroutine where it is first used. Not included here are temporary variables whose meaning is clear from their place in the program. In Section 3 a complete list of subroutines (including also those which are not used in the present version of the program) is given. Also the "tree" of the program is shown. In Section 4 the input file (hardware data) are explained in detail and an input file is given.

In Section 5. The role and the functioning of each subroutine in the program is explained. Finally, in Section 6 a few examples of output and the stack profile is shown.

# 2. DICTIONARY OF VARIABLE NAMES

(In parenthesis indicated where first used)

| | |
|---|---|
| AI | $i$ (imaginary unit) |
| AN | The current value of the harmonic number, used during the calculation in the subroutines where the response is computed (see the TRFFOL branch in the tree (Section 3) |
| AN1, AN2 | Bottom and top harmonic numbers, equal to $f_{min}/f_0$ and $f_{max}/f_0$ (INITL) |
| BP1 | $\frac{1}{|\eta_0|T^3}$ (START,INITL) |
| BP2 | $\frac{1}{|\eta_0|T^3}(\eta_0 - 2\frac{\eta_1}{\eta_0})$ (START,INITL) |
| BPBAR | $\beta$ (INITL) |
| BSQE | $\beta^2 E$ (INITL) |
| CF(3) | (START) |
| CD1(3) | (START) |
| CD2(3) | (START) |
| D1 | Coefficient in Fokker-Planck equation (AUX) |
| D2 | Coefficient in Fokker-Planck equation (AUX) |
| D1ST(1000) | D1/EBIN (FCN) |
| D2ST(1000) | D2/2*EBIN (FCN) |
| DAN(2) | (AN2–AN1)/NHAR for each system, distance between NHAR chosen harmonics in the interval (AN1,AN2) |
| DEINJ | Half width of injected beam |
| DFFDE | $\eta_0 + 2\eta_1 E$ (START) |
| DID(50) | (TRFFOL) |
| DPS | =Y(I+1)-Y(I), corresponds to $d\Psi$ |
| DSC | Intrabeam scattering contribution to D1/particle [eV$^2$/c$^2$s] (BLOCK DATA) |
| DSCN | $D_0$ coefficient in Fokker–Planck equation, = DSC$\times \int \Psi(E)dE$. (FCN) |
| EBIN | Size of energy bin [eV] (BLOCK DATA) |
| EE | Proton charge [C] (BLOCK DATA) |
| EINJ | Mean energy of injected pulse (BLOCK DATA) |
| EKIN | Nominal $\bar{p}$ kinetic energy in Accumulator (BLOCK DATA) |
| EMIN | Lower energy aperture (BLOCK DATA) |
| EMPBAR | Proton mass [eV] |
| ENINJ | Number of injected particles (5 mA/hour) (BLOCK DATA) |
| EPBAR | $\bar{p}$ energy (=EKIN + EMPBAR) (INITL) |
| ETA0 | $(\frac{1}{\gamma_T^2} - \frac{1}{\gamma^2})/\beta^2 E_{\bar{p}}$ (START, INITL) – *Note that this differs from the conventional def. of $\eta$* |
| ETA1 | $2/E_{\bar{p}}^2\gamma^2\beta^2$ (START, INITL) |
| F | Coefficient in Fokker-Planck equation (AUX) |
| FB(1000,5,3) | Beam feedback |
| FD1 | Arbitrary factor for D1 term (default 1) (BLOCK DATA) |
| FDEC1 | Factor for determining time step size in FCN |
| FLOSS | Number of particles lost between last and current injection (BLOCK DATA) |
| FLU(I) | *Flux* $=(D1+D2*\Psi)\frac{\Delta}{J}\Psi + F_{\frac{3}{}}\Psi$, *this is flux from lower toward higher energy* $\Psi(I)-\Psi(I+1)$ |

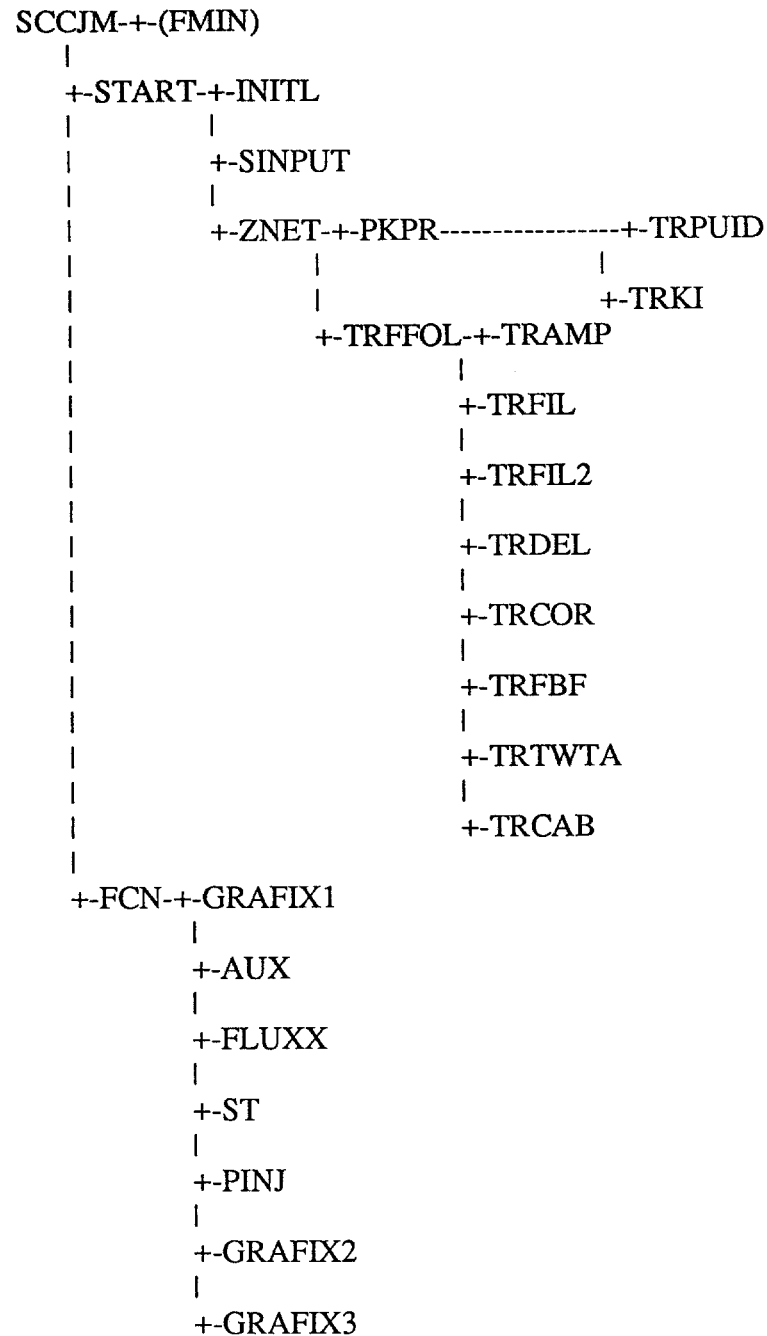| | |
|---|---|
| FMAX(2) | High end of frequency band for each system (BLOCK DATA) |
| FMIN(2) | Low end of frequency band for each system (BLOCK DATA) |
| FREQ | $(1 + \eta_0 E + \eta_1 E^2)/T$ (START, INITL) |
| GAMMA | $\gamma$ (INITL) |
| GAMMAT(3) | GAMMA(1)=$1/\gamma_T^2$, GAMMA(2)=GAMMA(3)=0 $\Leftarrow$ |
| GD(100,10,2) | Gain (ZNET, AUX) |
| GSCALE | (FCN, AUX) |
| H(8) | Step sizes in the integration of Fokker–Planck equation (FCN) |
| IFIN | (BLOCK DATA) |
| ISYS(3) | (START) |
| ITEM | The number assigned to each device in the chain from pickup to kicker (SINPUT) |
| ITF,ITFS | Parameters that determine step size in the integration of Fokker-Planck equation (BLOCK DATA) |
| ITYP | Type of element. 3, 4, 5, 6, 10, 11, 12, 13, corresponds to amplifier, notch filter, double notch filter, mixing between pickup and kicker, correlator, feedback filter, TWT amplifier, cable, respectively (PKPR) |
| KT((3) | (START) |
| LLBIN | Scale factor for the bin size (BLOCK DATA) |
| MHAR | Maximum number of harmonics (BLOCK DATA) (does not appear in the program) |
| MPU | Maximum number of pickups (BLOCK DATA) (does not appear in the program) |
| NBINJ | Bin number of center of injected pulse, = (EINJ–EMIN)/EBIN + 1.5 1 adds one bin and 0.5 does proper truncation to integers (START) |
| NCB | Dividing line between slow changing and fast changing parts of beam (START) |
| NH(5,2) | The harmonic numbers at which calculation is done, equals the bottom harmonic number plus the equally spaced numbers DAN(I) (INITL) |
| NHAR | Number of harmonics treated (BLOCK DATA) |
| NINJ | Number of injections to run, computed in START from given NTIME and TINJ, must be at least 5, otherwise a DO loop in FCN does not reach end (START) |
| NINJ1 | Lower bin of injected pulse, =(EINJ–DEINJ–EMIN)/EBIN + 1.5 (START) |
| NINJ2 | Upper bin of injected pulse, = 2*NBINJ-NINJ1 (START) |
| NITEM | Number of hardware items in the system (BLOCK DATA) |
| NLST | Number of steps in a cycle |
| NRF1 | (START) |
| NRF2 | (START) |
| NSYS | Number of different systems (=2) (BLOCK DATA) |
| NSTA | Total number of bins (START) |
| NSTA1 | NSTA–1 (INITL) |
| NTIME(3) | Total time in format (h, m, s) (BLOCK DATA) |
| PAVE(2) | Power in the two systems |
| PI | $\pi$ (INITL) |
| PI2 | $2\pi$ (INITL) |
| PONOS(2) | Thermal noise power, proportional to $kTg^2$ ($g$ gain) (ZNET) |
| POSLE(1000) | Low frequency contribution to Schottky noise power |

POSHE(1000) High frequency contribution to Schottky noise power

PPBAR $\quad\quad\quad\bar{p}$ momentum

PPT(1000,10,3) Beam feedback coeff. $\times$ PU–kicker mixing $\times$ PU response / harmonic number (TRPUID, FCN, TRKI, TRKIF, ZNET)

PS2 $\quad\quad\quad$ =Y(I+1)+Y(I), corresponds to $2\Psi(E)$

PSI1 $\quad\quad\quad$ Injection density $[(eV/c^2)^{-1}]$ (BLOCK DATA)

PWAY $\quad\quad\quad$ Injection direction (1=from high energy end, 2=from low energy end) (BLOCK DATA)

RADDEG $\quad\quad\quad$ $\pi/180$ (INITL)

RES(50) $\quad\quad\quad$ (TRFFOL)

RESP $\quad\quad\quad$ Variable describing the response of any element in the chain from the pickup to the kicker, computed in TRAMP, TRFIL, TRFIL2, TRDEL, TRCOR, TRFBF, TRTWTA, TRCAB and returned to TRFFOL where the GAIN is computed

STBMAX $\quad\quad\quad$ Stability limit in the feedback formula (BLOCK DATA,FCN)

T $\quad\quad\quad$ Revolution period (in seconds) (BLOCK DATA)

TIMST $\quad\quad\quad$ Starting time for the integration

TINJ $\quad\quad\quad$ Time between injections (in seconds) (BLOCK DATA)

TRF(1000,10,3) Transfer function (ZNET), computed in TRPUID. First index bin number, second harmonic number, third PU number

TRFF $\quad\quad\quad$ Proportional to TRF (AUX)

XBIN(1000) $\quad\quad\quad$ The energy variable, defined and used solely for the purpose of graphics (FCN)

Y(1000) $\quad\quad\quad$ $\Psi(i)$, number of particles in the $i$-th energy bin

YLOG(1000) $\quad\quad\quad$ $\log_{10}$ Y, defined and used solely for the purpose of graphics (FCN)

# 3.LIST OF SUBROUTINES AND THE TREE

Subroutines are listed in order they appear in the program.  The subroutines which have at some point been included but are presently not being used are marked 'not used'.  They have been commented out and not removed.

SCCJM
BLOCK DATA
START
FCN
GRAPHINIT
PLOTD1
PLOTD2
PLOT D2PSI
PLOT D1D2RAT
PLOT F
PLOT PSI
PLOT FLUX
FREQ(E)
CINT(A,C) (not used)
INITL
SINPUT
PKPR(JI)
TRAMP
TRCAB
TRCOR
TRDEL
TRFBF
TRFFOL(JN,GAIN,E,AN,JK)
TRGPC
TRKI
TRTWTA
ZNET
AUX
FLUXX
PINJ
ST            PINJ2
TRFIL (not used)
TRFIL2 (not used)
TRKIF (not used)
TRPUF (not used)
TRPUID
TRPUL (not used)

The tree of the program is as follows:

```
SCCJM-+-(FMIN)
      |
      +-START-+-INITL
      |       |
      |       +-SINPUT
      |       |
      |       +-ZNET-+-PKPR----------------+-TRPUID
      |             |                      |
      |             |                      +-TRKI
      |             +-TRFFOL-+-TRAMP
      |                      |
      |                      +-TRFIL
      |                      |
      |                      +-TRFIL2
      |                      |
      |                      +-TRDEL
      |                      |
      |                      +-TRCOR
      |                      |
      |                      +-TRFBF
      |                      |
      |                      +-TRTWTA
      |                      |
      |                      +-TRCAB
      |
      +-FCN-+-GRAFIX1
            |
            +-AUX
            |
            +-FLUXX
            |
            +-ST
            |
            +-PINJ
            |
            +-GRAFIX2
            |
            +-GRAFIX3
```

In what follows we shall describe the role of each subroutine in the tree. In the Appendix we describe briefly the 'not used' subroutines.

## 4. THE INPUT FILE

At present there are two input files, for the 1–2 and 2–4 GHz stack tail system, respectively. Their structure is identical and we show here the 1–2 GHz file.

The data are organized in groups, one group for each device. The first number in each group is an integer, stored in IWDS, which indicates how many data are in the group. The first 7 elements that follow IWDS are integers, the rest IWDS–8 reals. The integers are stored in IWS(2)...IWS(8), the reals in WS(9)...WS(IWDS+1). These variables have the following meanings:

IWS(2)          Item number
IWS(3)          System number, 1 refers to stack tail and 2 to core cooling system
IWS(4,5)        Upstream links, *i.e.* Item numbers of the upstream elements.
                Zero means non existing link, *e. g.* the pickup has 0 0 since there
                is no upstream element.
IWS(6,7)        Same as IWS(4,5) for downstream links.
IWS(8)          Item type, ITYP. 3 is amplifier, 6 mixing (delay), 7 kicker, 9 pickup,
                10 filter, 12 TWT amplifier, 13 correlator cable.
WS(9)           Depends on the device. For notch filters it is the energy of the notch
                in MeV relative to central energy, for amplifier it is gain.
WS(10)          Depends on the device. Phase for the amplifier.
WS(11)          Depends on the device. The noise figure of the device in dB (amplifier),
                number of entries in the table of gains and phases for TWT amplifier (here 71)
                pickup to kicker distance as fraction of circumference (pickup).
WS(12)          Depends on the device. For pickup, contribution to $\eta$ from pickup
                to kicker, for amplifier phase change from group delay
WS(14)          Width of pickup plates
WS(15)          Gap between pickup plates

For the TWT amplifiers, WS(12) through WS(223) are tables of gain (second column) and phase (third column) *vs.* frequency in GHz (first column) in steps of 20 MHz. WS(11) indicates how many data points there are in the table.

Here is the input file SYS12 DATA (the names of the devices and the meaning of the data can be deduced from the above list and are not given here in full detail):

**Stack tail system (IWS(3)=1)**
Pickup:
16 1 1 0 0 2 0 9
16. 160.0 0.5 0.5 .107E-6 4.5 3.0 -1. .1471
Amplifier:
11 2 1 1 0 3 0 3
0.80E+5 2.70 1.0 0.0
Delay:
8 3 1 2 0 4 0 6
2.00

Filter:
12 4 1 3 0 5 0 10
-57. 1.E-6 .97 0.444E-2 0.889
Combiner:
11 5 1 -4 -14 6 9 3
0.5 0. 0. 0.
Phase/gain adjustment:
11 6 1 5 0 7 0 3
0.63 0.0 0.0 0.00
Correlator cable:
8 7 1 6 0 8 0 13
-63.
TWT (71 points of frequency, gain and phase follow):
223 8 1 7 0 10 0 12
0. 4000. 71.
.80 37.20 22.71
.82 38.64 19.60
.84 39.78 15.83
.86 37.77 13.70
.88 38.53 19.61
.90 40.95 12.96
.92 40.47 2.65
.94 42.40 .77
.96 42.75 2.33
.98 42.25 6.01
1.00 43.71 8.01
1.02 43.22 -.48
1.04 43.05 -.08
1.06 44.14 -3.11
1.08 45.14 -17.06
1.10 46.32 10.62
1.12 46.38 -5.96
1.14 44.48 10.26
1.16 46.49 -6.13
1.18 44.54 3.75
1.20 47.02 -7.82
1.22 46.02 -5.05
1.24 47.72 5.74
1.26 46.63 -6.87
1.28 46.32 19.31
1.30 44.04 -20.13
1.32 46.94 7.11
1.34 47.09 -3.39
1.36 47.32 -.72

1.38 48.06 -1.19
1.40 47.39 3.79
1.42 47.39 4.04
1.44 47.04 1.84
1.46 46.80 1.21
1.48 46.73 -3.56
1.50 47.03 -.85
1.52 47.49 -1.48
1.54 46.89 4.28
1.56 46.78 2.91
1.58 46.23 1.84
1.60 45.37 .96
1.62 45.86 .37
1.64 46.21 -5.11
1.66 46.01 -.28
1.68 45.83 -1.24
1.70 44.64 5.42
1.72 44.54 2.00
1.74 44.17 .11
1.76 43.81 -4.65
1.78 43.84 -3.98
1.80 43.87 -2.87
1.82 43.54 2.49
1.84 43.43 -.70
1.86 41.83 .45
1.88 41.31 2.06
1.90 41.26 -4.08
1.92 40.99 -5.45
1.94 41.20 -2.06
1.96 40.98 .49
1.98 39.82 6.11
2.00 39.78 2.70
2.02 38.40 2.07
2.04 38.04 .81
2.06 37.92 -.67
2.08 38.37 3.93
2.10 37.32 6.01
2.12 36.53 11.48
2.14 35.38 8.53
2.16 34.15 7.07
2.18 34.33 5.30
2.20 34.26 6.93

TWT (same as above):
223 9 1 5 0 10 0 12
0. 4000. 71.
.80 34.31 5.66
.82 34.84 11.13
.84 35.98 6.26
.86 35.50 6.23
.88 36.56 6.76
.90 36.54 8.12
.92 36.84 1.33
.94 38.46 -1.82
.96 38.85 -2.56
.98 39.11 -1.98
1.00 39.77 1.69
1.02 40.31 4.04
1.04 39.70 -2.44
1.06 40.49 -2.18
1.08 40.81 -3.39
1.10 41.02 -4.31
1.12 41.70 -2.74
1.14 42.21 -4.01
1.16 42.26 1.25
1.18 42.58 .44
1.20 42.10 2.64
1.22 42.29 .93
1.24 42.62 -.91
1.26 42.23 5.83
1.28 42.80 -8.57
1.30 43.24 -.32
1.32 43.78 -.02
1.34 43.31 1.13
1.36 43.22 3.74
1.38 43.09 1.59
1.40 42.68 2.20
1.42 42.67 -.94
1.44 42.94 -1.52
1.46 42.65 -6.03
1.48 43.11 .83
1.50 43.11 1.56
1.52 42.58 1.08
1.54 42.31 3.68
1.56 42.12 .07
1.58 41.31 -3.04
1.60 41.09 -3.35

1.62 41.42 -4.14
1.64 41.56 -3.53
1.66 41.39 1.91
1.68 41.34 -.84
1.70 40.41 .45
1.72 39.74 .27
1.74 39.47 3.13
1.76 39.19 -2.87
1.78 38.99 -.93
1.80 39.11 -7.24
1.82 38.98 -.59
1.84 38.40 -2.59
1.86 37.73 6.17
1.88 37.22 1.19
1.90 36.80 6.58
1.92 35.79 -2.86
1.94 35.20 7.57
1.96 35.48 -5.63
1.98 35.41 5.35
2.00 34.75 -2.49
2.02 34.64 16.65
2.04 33.50 2.18
2.06 32.70 22.69
2.08 32.95 7.69
2.10 31.73 12.38
2.12 31.79 -2.14
2.14 31.15 12.94
2.16 31.51 7.71
2.18 30.65 22.39
2.20 29.80 15.17
Kicker:
9 10 1 8 9 0 0 7
89.40 1.69
Pickup:
16 11 1 0 0 12 0 9
-1. 80.0 0.5 0.5 .107E-6 4.5 3.0 -23. .0853
Amplifier:
11 12 1 11 0 13 0 3
0.40E+5 0.60 1.0 0.0
Delay:
8 13 1 12 0 14 0 6
-3.0

Filter:

12 14 1 13 0 5 0 10

-64. 1.E-6 .97 0.444E-2 0.889

**Core cooling system** (IWS(3)=2)

Pickup:

16 21 2 0 0 22 0 9

-35. 69.28 0.5 0.5 .107E-6 2.5 3.0 -87. 1.0

Amplifier:

11 22 2 21 0 23 0 3

1.80E+5 3.14 4.0 0.0

Delay:

8 23 2 22 0 24 0 6

-30.5

Kicker:

9 24 2 23 0 0 0 7

33.64 1.33

## 5. FUNCTIONING OF THE PROGRAM

All file names are CMS names, *e.g.* COMMSTOC FORTRAN and have to be appropriately changed for use under VMS or UNIX.

### 1. SCCJM

This program opens the correct input file containing hardware data. At present there are two such files, SYS12 DATA and SYS24 DATA for the 1–2 and 2–4 GHz stack tail cooling system, respectively.

Next the output file STACK DATA is opened and START and FCN are called. After that SCCJM ends the execution of the program.

### 2. START

The number of injections NINJ is computed from the given time of stacking NTIME. The energy bin is computed from the default value of 1.4 MeV and the scale factor LLBIN (default = 1) in order to accomodate cases where the results are unstable. LLBIN will also determine the total number of bins NSTA and the variable NCB.

### 3. INITL

This subroutine computes and stores the basic constants $i$, $\pi$, $E_{\overline{p}}$, $p_{\overline{p}}$, $\beta$, $\beta^2 E$, $\gamma$, $\eta_0$, $\eta_1$. $\gamma_T$ is given as a function of energy:

$$\frac{1}{\gamma_T^2} = \text{GAMMAT}(1) + \text{GAMMAT}(2) \times E_{\overline{p}}.$$

From the given revolution period and FMIN and FMAX for each system their bottom and top harmonic numbers AN1 and AN2 are computed. In the range AN2–AN1 (for each system) NHAR (default = 5) equally spaced numbers are stored in DAN(I). The corresponding NHAR harmonic numbers are stored in NH(I,J) with dimension (NHAR,NSYS).

### 4. SINPUT

The hardware data from the selected input file (SYS12 DATA or SYS24 DATA) are read.

### 5. ZNET

Basically contains several nested do loops going through all the devices (ITEM values) at the NHAR chosen harmonics and calls TRFFOL to obtain their response. Also computes the noise level. KT(NPU) is $k$T for a given pickup system NPU. There are 3 PU systems, 1 and 2 are stack tail, 3 is core. T=300 °K for the transmission line, 80 °K for the cryogenic amplifier, $k$ Boltzmann constant. KT is determined from the noise figure read from the input file.

### 6. PKPR(JI)

Controls processing of pickup and kicker type elements. If the element is a kicker the subroutine TRKI is called, if it is a pickup TRPUID is called.

### 7. TRPUID

Calculates the pickup response. The transfer function is

$$-\frac{1}{\pi}e^{i\phi}\sqrt{ZN_{pu}}\sin\frac{\pi}{2}\frac{h}{<h>}\left(\arctan\frac{\sinh(\pi w/2d)}{\cosh(\pi x/d)} - B\arctan\frac{\sinh(\pi w/2d)}{\cosh(\pi x_c/d)}\right),$$

14

where $w$ is the width of the pickup plates, $d$ the gap between plates, $x$ the distance from the pickup ($x_c$ for the compensating pickup), $N$ the number of pickups, $Z$ their characteristic impedance, $h$ the harmonic number and $< h >$ the harmonic number corresponding to mid band. The calculation is done for NHAR Schottky harmonics.

8. TRKI

Calculates the kicker response with constant sensitivity (*i.e.* independent of frequency). The sensitivity is

$$S = B \sin(\frac{\pi}{2} \frac{h}{<h>}),$$

where $B$ is a measured coefficient read from input file, $< h >$ the harmonic number corresponding to mid band. The transfer function is multiplied by $S$ and returned.

9. TRFFOL(JN,GAIN,E,AN,JK)

Constructs the gain function from the pickup to the kicker. The elements are modeled in the corresponding subroutines which are called according to the value of ITYP, the element number. ITYP=3 corresponds to amplifier and TRAMP is called, 4 is notch filter and TRFIL is called, 5 is the double notch filter (TRFIL2), 6 is mixing between pickup and kicker (TRDEL), 10 correlator (TRCOR), 11 feedback filter (TRFBF), 12 TWT amplifier (TRTWTA) and 13 cable (TRCAB). The returned variable from all subroutines is RESP. GAIN is initialized to RES(IU1) and multiplied by RESP to obtain total GAIN. JN is the pointer, E the energy, AN the harmonic number, JK the pointer.

10. TRAMP

The response of the amplifier is $ge^{i\phi}$ where $g$ and $\phi$ are read from the input file. Phase change from group delay is $\phi_d = 2\pi h f(E) \times measured\ coefficient \times 10^{-9}$, where $h$ is the harmonic number, $f$ the frequency corresponding to the energy E. The total response is

$$RESP = ge^{i(\phi+\phi_d)}$$

11. TRFIL

Calculates the response of an ideal notch filter. The response is

$$RESP = \frac{1}{B} \frac{1}{\frac{e^{f_L+i(f+f_L)}+e^{-f_L-i(f+f_L)}}{e^{f_L+i(f+f_L)}-e^{-f_L-i(f+f_L)}} + \frac{1}{B}},$$

where $B$ is a measured coefficient, $f_L = \sqrt{\frac{h}{<h>}}B$,

$$f = h\pi \frac{1 + \eta_0 E + \eta_1 E^2}{1 + \eta_0 E_n + \eta_1 E_n^2}, \tag{$*$}$$

and $E_n$ the energy of the notch.

12. TRFIL2

Calculates the response of an ideal double notch filter (two shorted lines in parallel). The impedance of each line is

$$Z = \frac{1}{B} \frac{e^{f_L+i(f+f_L)} - e^{-f_L-i(f+f_L)}}{e^{f_L+i(f+f_L)} + e^{-f_L-i(f+f_L)}},$$

15

where $f$ is computed according to Eq. (*). $B$ is read from the input file. Since they are connected in parallel the impedance is

$$Z = \frac{Z_1 Z_2}{Z_1 + Z_2}$$

and the response is

$$RESP = \frac{Z}{1 + Z}.$$

13. TRDEL

Calculates the mixing between the pickup and the kicker. The 'response' of mixing is

$$RESP = e^{i\phi},$$

where $\phi = 2\pi h(\eta_0 E_s + \eta_1 E_s^2)$ and $E_s$ a measured energy [MeV].

14. TRCOR

Calculates the correlator response.

$$RESP = \frac{1}{2}(1 - T \exp(-\phi\sqrt{f} - i2\pi f/f_n),$$

where $f = hf(E)$, $f_n = f(E_n)$ ($E_n$ the characteristic energy). The values of $T$, $\phi$ and $E_n$ are read from the input file.

15. TRFBF

Calculates the feedback filter response.

$$RESP = \frac{1}{1 + Ae^{-i2\pi f/f_n}},$$

where $f = hf(E)$, $f_n = f(E_n)$, and the values of $A$ and $E_n$ are read from the input file.

16. TRTWTA

Calculates the (complex) response of the TWT amplifier. The frequencies $f(I)$ and the absolute value of the amplifier gain AMP and the phase PHI are read from the input data and the real and imaginary parts of gain $g_R$ and $g_I$ (=AMP*cos(PHI) and AMP*sin(PHI), respectively) are obtained as a function of frequency. Finally, the response is computed as a function of frequency:

$$RESP = \frac{(f - f(I))(g_R(I+1) - g_R(I))}{f(I+1) - f(I)} + g_R(I)$$

$$+i\Big[\frac{(f - f(I))(g_I(I+1) - g_I(I))}{f(I+1) - f(I)} + g_I(I)\Big].$$

17. TRCAB

Calculates the response of an ideal cable.

$$RESP = -e^{-i2\pi f/f_n},$$

where $f_n$ is the frequency corresponding to the characteristic energy (read from the input data).

18. FCN

Here the integration of the Fokker–Planck equation is done. The integration is done for the required number of injections (NINJ). The step size for the integration is

$$H(N) = \frac{T_{inj}}{E_{bin}} 2^{N-\text{ITF}-8}, \quad (N = 1, \ldots, 8)$$

ITF is a constant assigned a value in BLOCK DATA to control size of $H(N)$. $H(N)$ is changed during the integration, the nested DO loops are there to allow for different step sizes. NN is the group number (in COMMON). FLUXX calculates the flux from the energy distribution and ST calculates the new density from the flux gradient. The intrabeam scattering cofficient DSCN is computed for each energy bin from the current particle density $\Psi(E)$ and the given (BLOCK DATA) coefficient DSC:

$$DSCN = DSC \times \Psi(E)\Delta E,$$

where $\Delta E$ is the bin size. The subroutine AUX is called and the cofficients $F$, $D_1$ and $D_2$ of the Fokker-Planck equation obtained.

19. GRAPHINIT

Initializes the graphics package HIGZ.

20. AUX

Calculates and returns the coefficients D1, D2 and F of the Fokker-Planck equation. Also computes the Schottky noise power POSCH. (Comment on GSQ,DEN,TRFF)

21. FLUXX

Calculates flux according to the formula

$$\Phi(E) = -F(E)\Psi(E) + \big(D_0 + D_1(E) + D_2(E)\Psi(E)\big)\frac{\partial \Psi}{\partial E}.$$

Here, $D_0$ and $D_1$ are added into $D_1(E)$ and the formula used in the computation is

$$\text{FLU(I)} = \big(\text{D1(I)} + \text{D2(I)} \times \Psi(I)\big)\frac{\Delta \Psi}{\text{EBIN}} - \text{F(I)}\Psi(I).$$

22. ST

Redistributes density. Calculates the new density from the flux gradient.

23. PINJ

Injects individual pulses of antiprotons into the accumulator. PINJ2

24. PLOT D1

Plots the coefficient $D_1$ of FPE as a function of energy.

25. PLOT D2

Plots the coefficient $D_2$ of FPE as a function of energy.

26. PLOT D2PSI

Plots the product $D_2\Psi$ as a function of energy.

17

27. PLOT D1D2RAT
Plots the ratio $D_1/D_2$ as a function of energy.
28. PLOT F
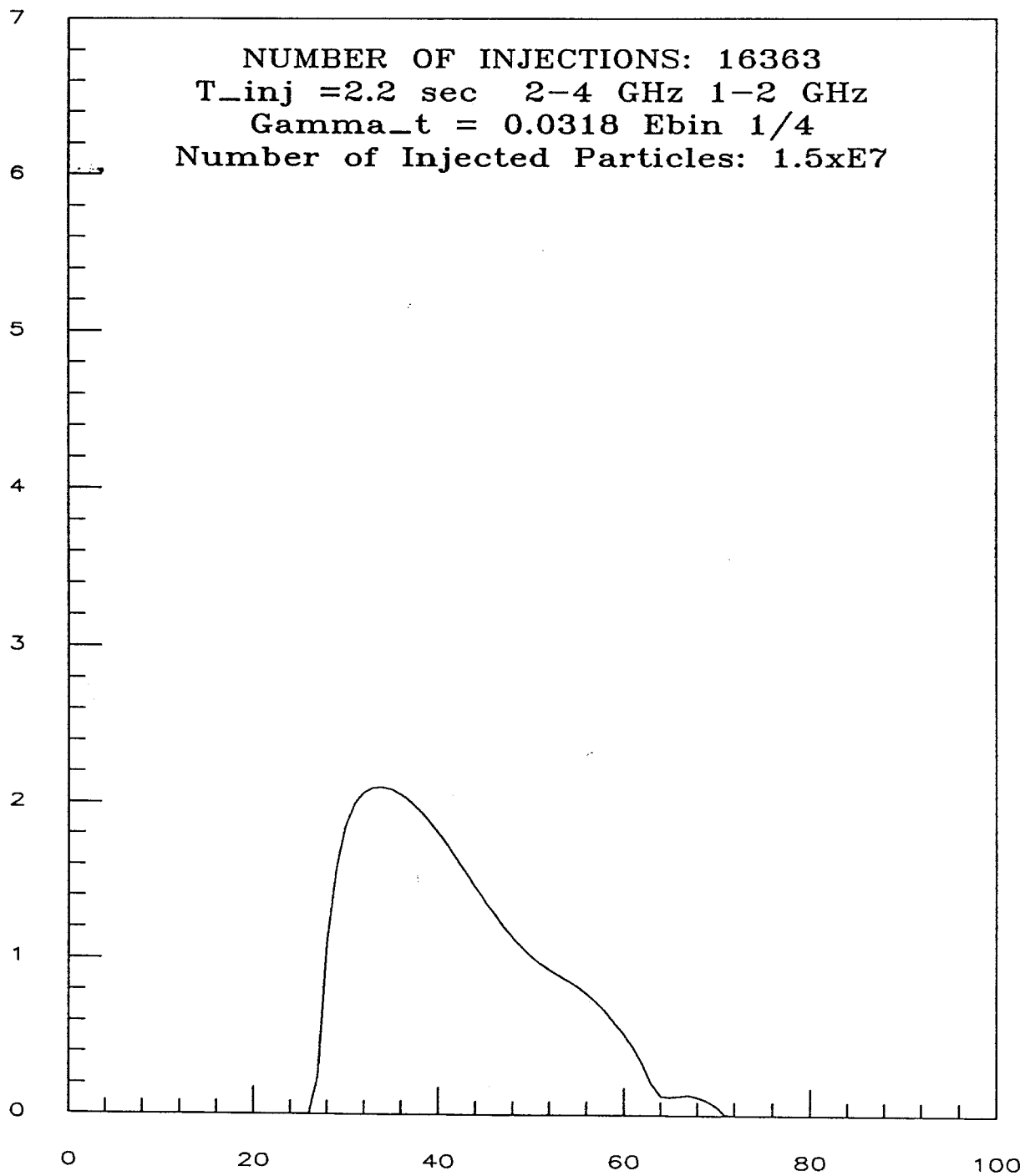Plots the coefficient $F$ of FPE as a function of energy.
29. PLOT PSI
Initializes the plot of $\log_{10} \Psi(E)$ versus E which is done in FCN.
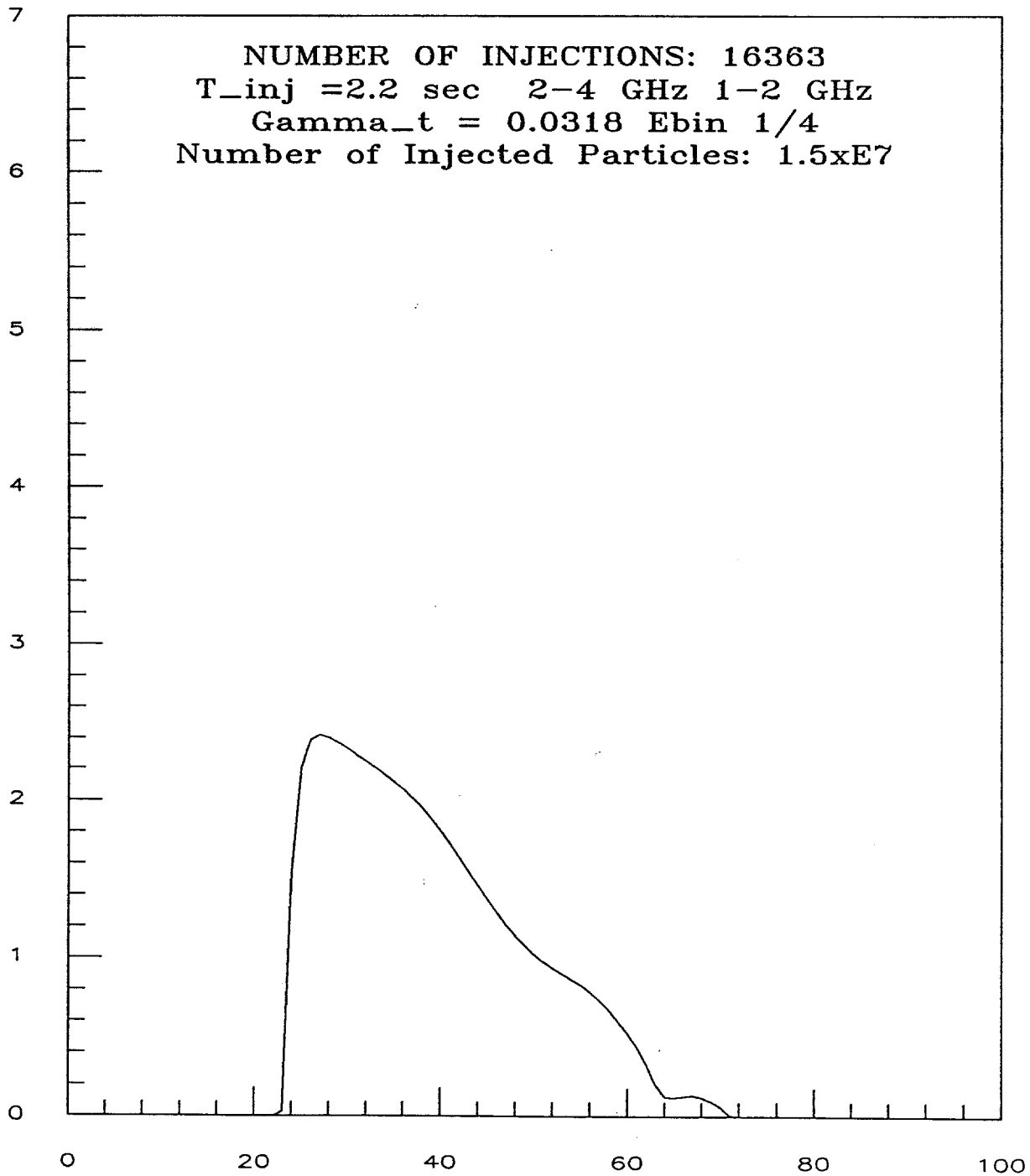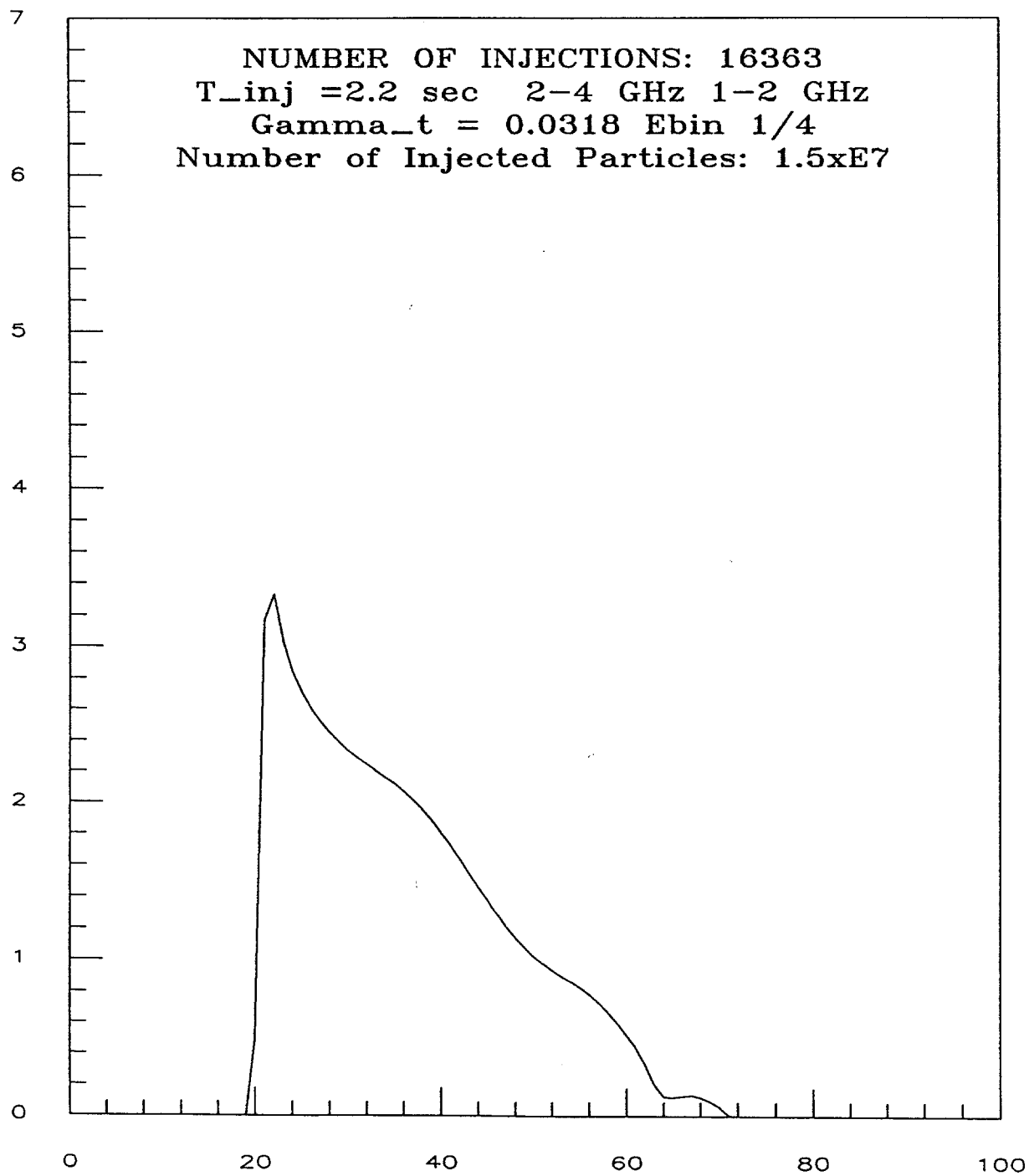30. PLOT FLUX
Plots flux as a function of energy.
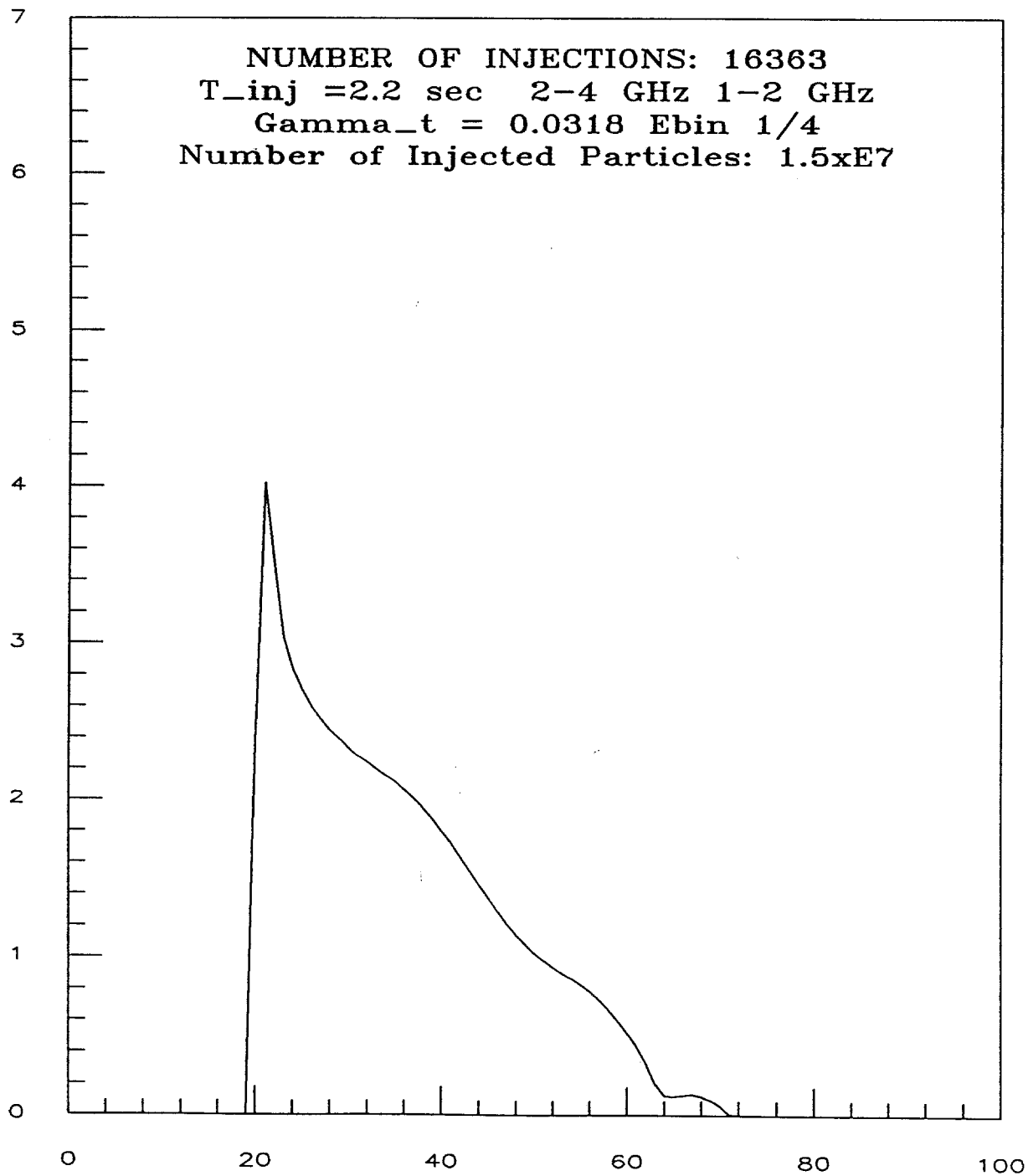
## 6. RESULTS

As an example of the use of the program, we show here the beam profile during 10 hr. of stacking. First the $D_1$ cofficient of the Fokker-Planck equation is shown as function of frequency. This particular example turns out to have the instability mentioned before (see Introduction) and required $E_{bin}$ of 1/4 the default size to solve. This is indicated in the pictures of the beam profile. The time sequence of beam profile is 10 s, 30 s, 1 m, 2 m, 5 m, 10 m, 30 m, 1 h, 2 h, 3 h, 4 h, 5 h, 6 h, 8 h and 10 h, respectively.
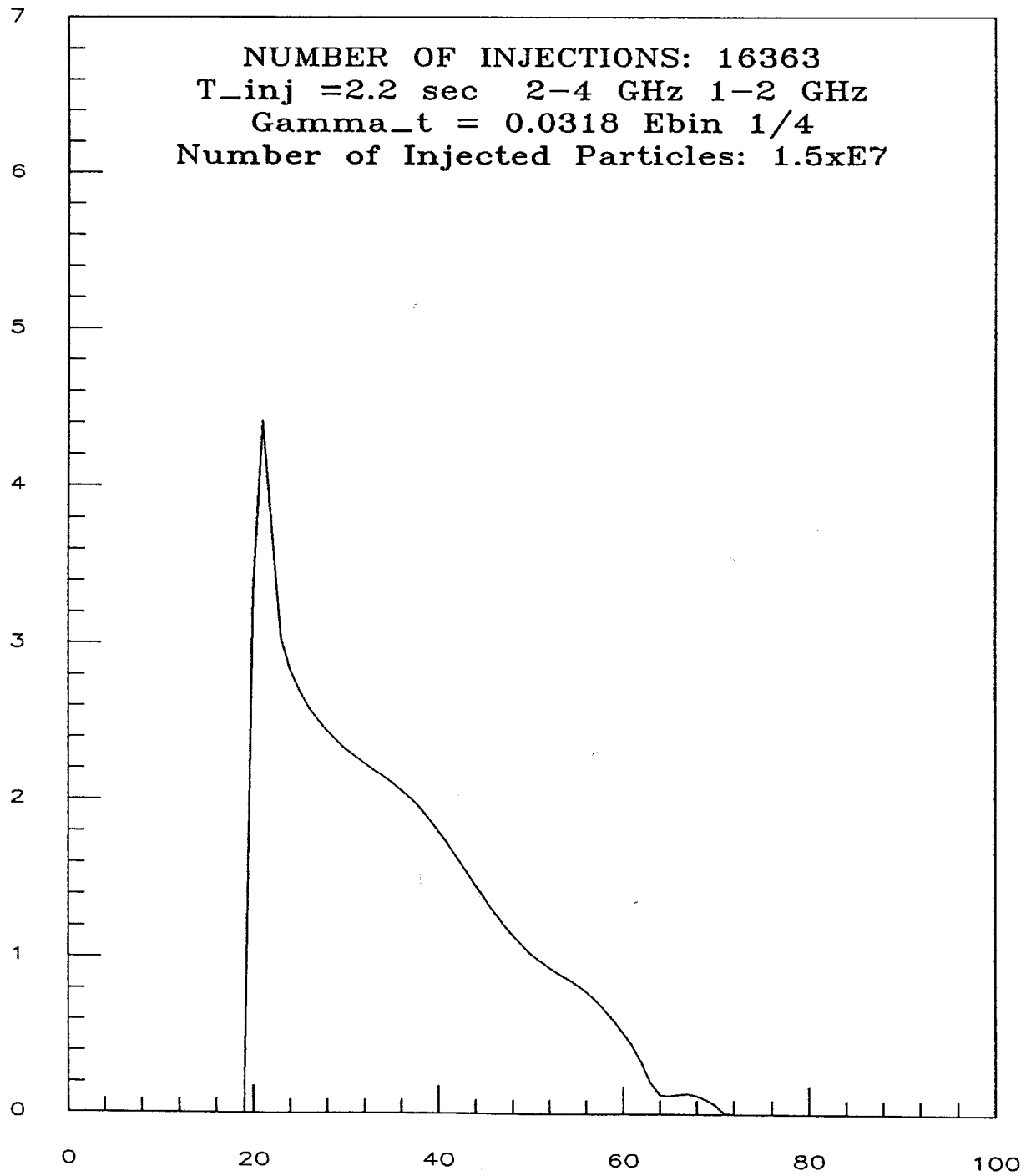
NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
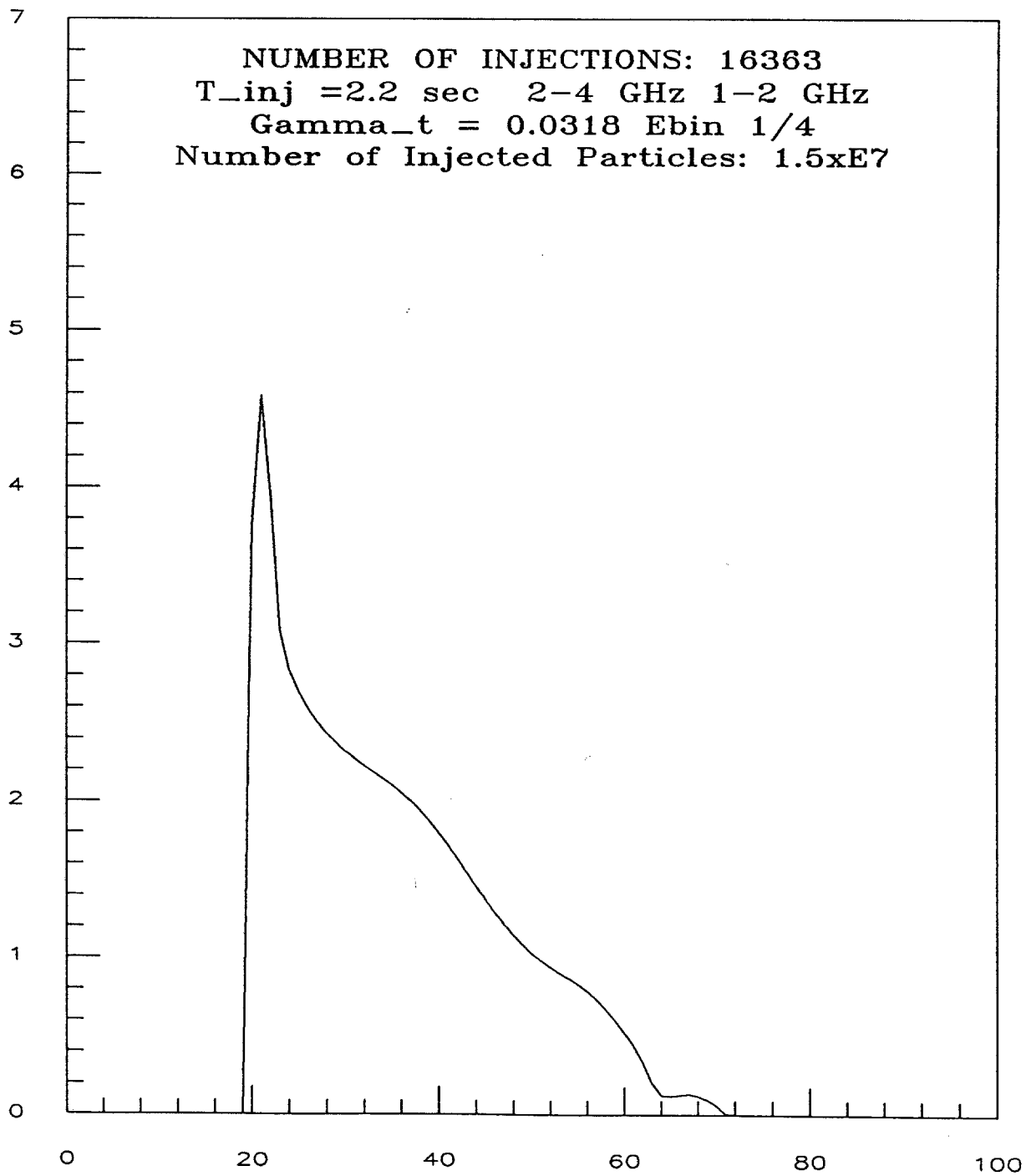Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj = 2.2 sec   2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec    2-4 GHz 1-2 GHz
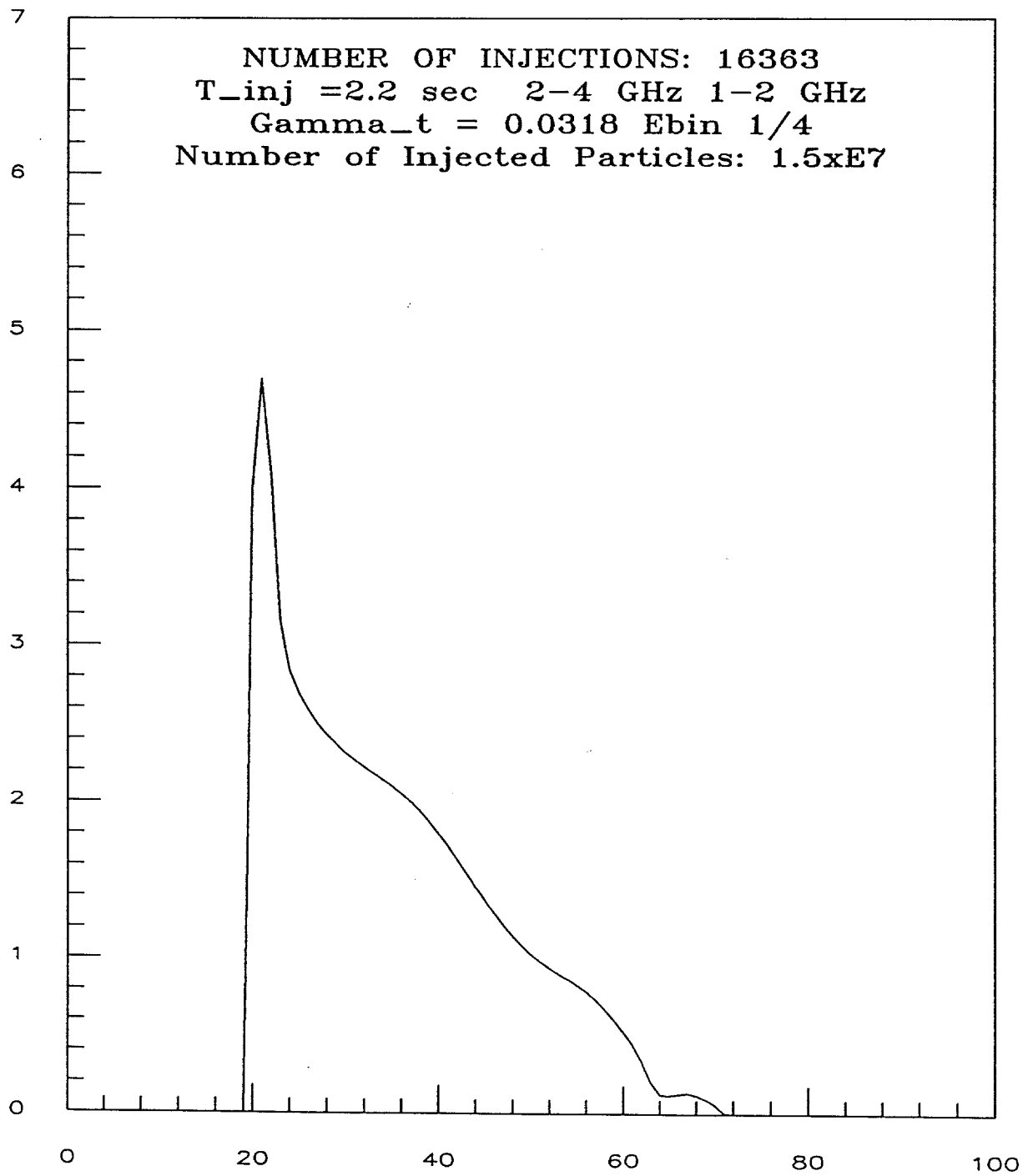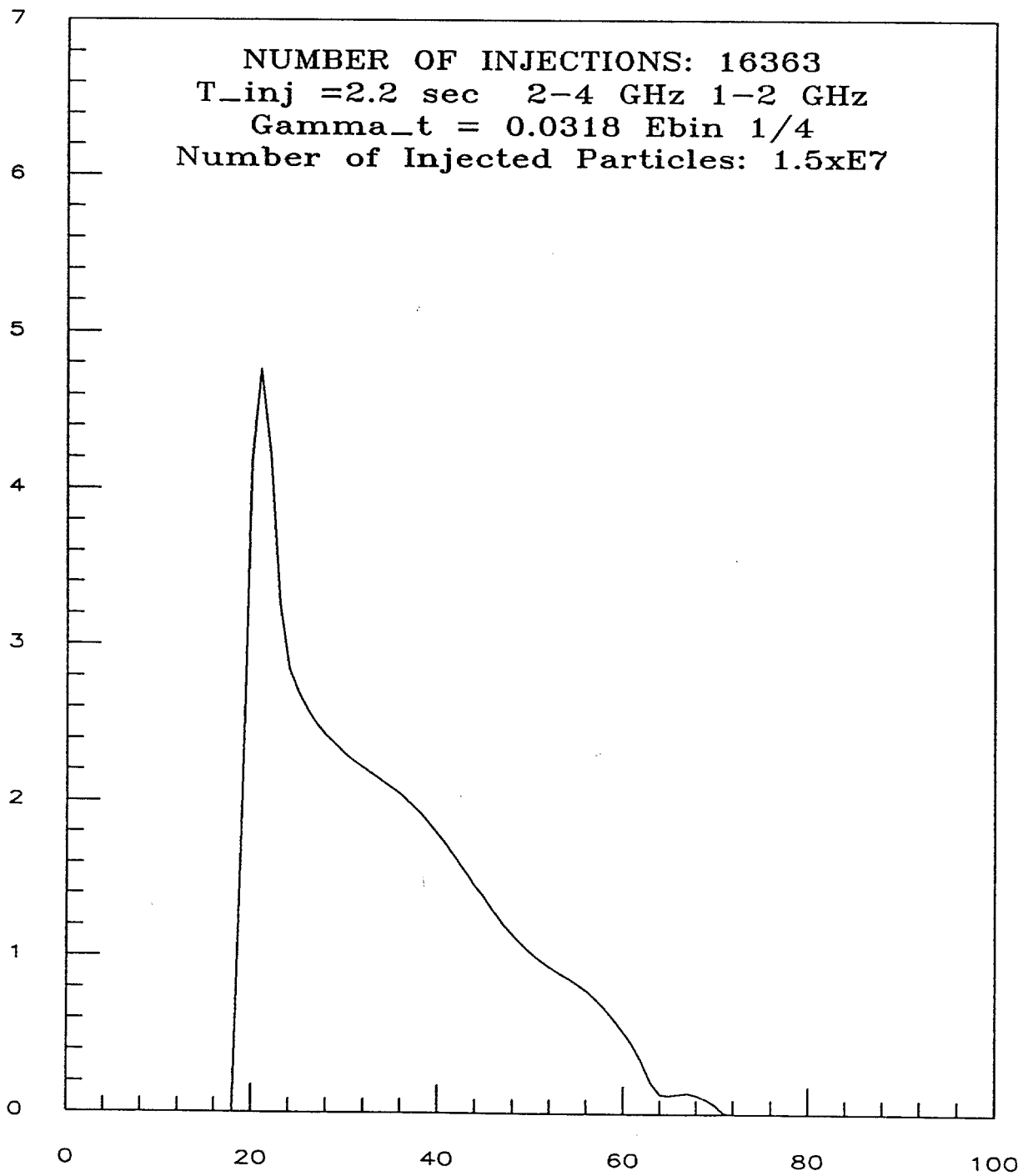Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec    2-4 GHz 1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4  GHz  1-2  GHz
Gamma_t  =  0.0318  Ebin  1/4
Number  of  Injected  Particles:  1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz 1-2 GHz
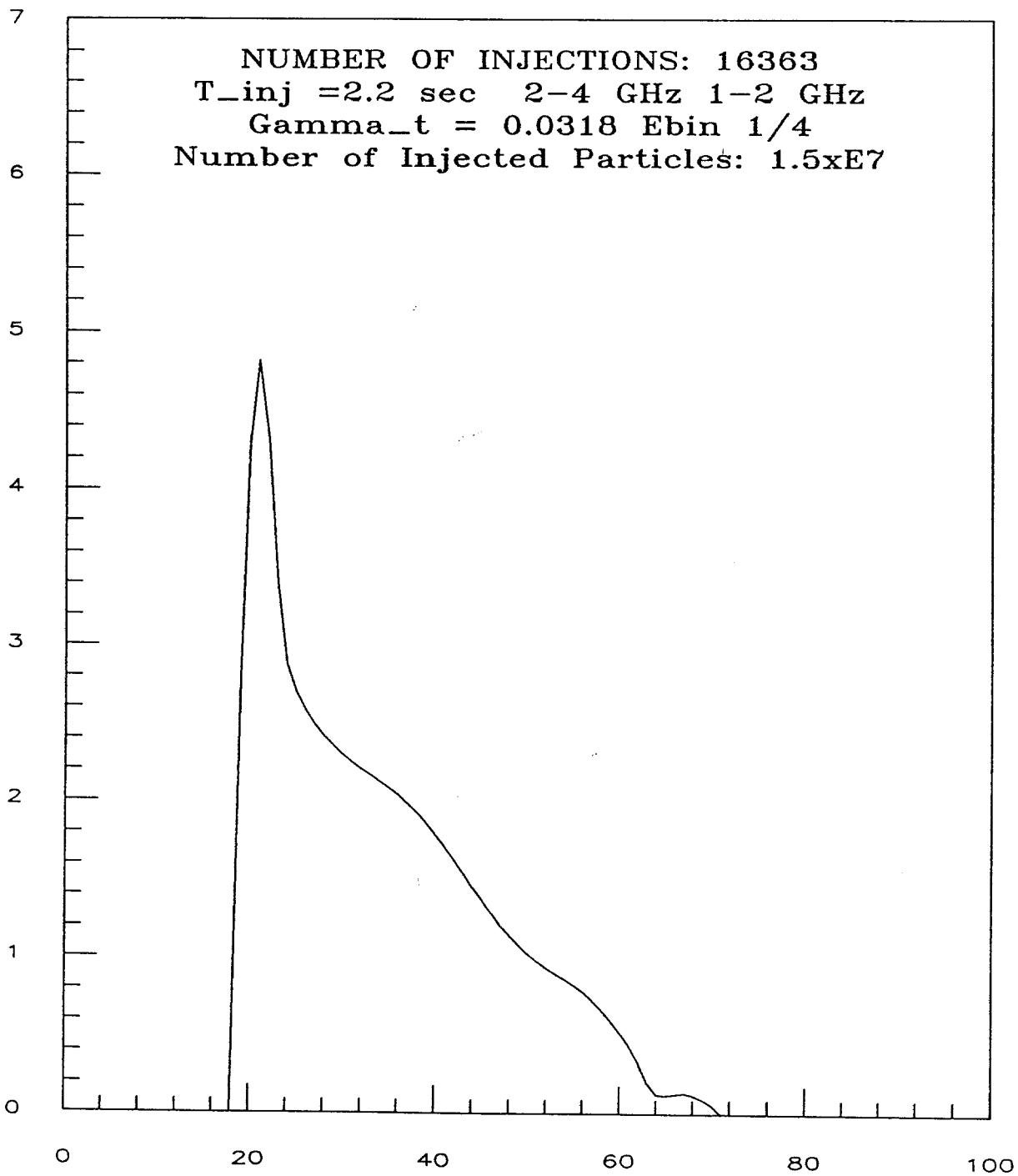Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2-4 GHz  1-2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7

NUMBER OF INJECTIONS: 16363
T_inj =2.2 sec   2−4 GHz 1−2 GHz
Gamma_t = 0.0318 Ebin 1/4
Number of Injected Particles: 1.5xE7